Teaching Digital Systems Design Through Computer EDA Tools And Breadboard Techniques

Guoping Wang
Department of Engineering
Indiana University Purdue University Fort Wayne
2101 E. Coliseum Blvd. Fort Wayne, IN 46805

Abstract

Digital Systems Design is a very important course in electrical and computer engineering programs. Integration of hardware description language such as Verilog or VHDL in the teaching has been proposed in the past. This paper describes the author's experience in integrating Xilinx ISE tools and FPGA/CPLD logic devices together with traditional breadboard techniques into the teaching of digital logic systems in Indiana University Purdue University Fort Wayne. The author's experience indicates that integrating EDA (Electronic Design Automation) tools and FPGA/CPLD devices together with breadboard lab sessions, instead of the early introduction of HDL, can better facilitate the learning of digital systems design.

Index term: -- *EDA, digital systems design, computer simulation*

Introduction

Digital hardware plays a dominant role in many electrical and computer engineering products today. The introductory course Digital Systems Design is a core requirement course for electrical and computer engineering students. Integrating HDL (Hardware Description Language), such as Verilog or VHDL into the teaching of logic system design has been proposed in the past[1-5]. However, for the early introduction of HDL, the instructor has to focus on explaining the important differences between HDL and other computer programming languages familiar to students. Students may focus upon the complex features of HDL instead of the fundamentals and principles of logic systems. If HDL is introduced at the same time as the logic system, then it is a big distraction from the teaching of these fundamental principles and practices of digital logic systems. These basic principles include Boolean algebra, Karnaugh maps, synchronous and asynchronous sequential circuits, to name a few. Because the schematic concepts and the digital logic blocks are the blueprints of digital logic systems, it is necessary to help students learn the basic principles first instead of jumping to VHDL directly. While it is agreed that students must learn about these digital logic functions and their implementations, whether to introduce HDL language too early into the teaching of digital system designs remains a controversy. The author's experience at IPFW (Indiana University Purdue University – Fort Wayne) shows that the schematic design using EDA tools and FPGA/CPLD implementation in the lab sessions, rather than HDL, can be introduced to students early on to facilitate their learning of digital logic systems.

ECE 270 -- Introduction to Digital System Design is a core course offered every semester at IPFW. This course extends over fourteen weeks with three hours of lectures per week and twelve laboratory sessions. Topics include combinational circuits (such as decoder, encoder, arithmetic functions, multiplexers) and sequential circuits (such as latches, flip-flops, counter, shift registers and sequential state machines). The main purposes of this course are (1) to teach students the basic concepts and functional blocks in digital logic design and (2) to illustrate the design and implementation procedure of digital logic systems (3) to troubleshoot and debug the digital logic systems from hands-on lab exercises. Extensive examples are introduced throughout this course. The different radix representations are

introduced first, followed by CMOS logic gates, static and dynamic characteristics, Boolean algebra, Karnaugh maps, combinational circuit analysis and practices, sequential circuit analysis and practices, and finally, sequential state machines. The textbook "Digital Design: Principles and Practices" [6] is adopted since there are lots of examples inside this book. It has a good supporting website for both students and the instructor[7].

The lab exercises are based on five experiments using bread-board, seven lab sessions using design in Xilinx ISE schematic capture, Modelsim simulator and XSA-100 FPGA boards. From the exercises on bread-board, students learn the debugging and troubleshooting techniques in the implementation and verification of logic systems. In the schematic design, simulation and verification using EDA and FPGA boards, students focus on the learning of logic block functions and the design of a moderately complex digital system using these digital logic blocks.

VHDL Teaching Experience in IPFW

ECE 357 – Introduction to VHDL is a junior level course intended for ECE students at IPFW. The main component of this course is the design of digital systems using VHDL. Topics covered include VHDL concurrent and sequential statements, signals and variables, state machine design, VHDL synthesis, simulation. Students had the opportunity to do hands-on projects on FPGA/CPLD boards. Because hardware description language is very different from traditional software programming languages like C, Matlab, the instructor found that a comprehensive understanding of digital logic system principles would help students much better in this course. For example, nearly two lecture sessions were spent to explain the difference between signals and variables. The instructor also spent two lectures to explain the concurrent and sequential statements and the corresponding digital logic circuits. Students also find it very confusing to design the sequential logic circuits with different control signals such as Clock enable, Set/Reset, Load, Output Enable, etc. Without prior knowledge of digital systems, it would be very difficult to learn VHDL. If VHDL is introduced at the same time as the digital logic systems, the 3 credit hours would not be enough to cover everything in sufficient detail; in the author's experience, at least 4 or 5 lecture credit hours would be required if both VHDL and digital systems design were covered in the teaching. The author would suggest that VHDL be introduced as a separate course at junior level after the introduction of digital systems design.

EDA Entry Tools Selection

Among digital systems design EDA tools, Altera MAX and Xilinx ISE tools are the two popular ones. Both of them have University Programs. The author chose Xilinx ISE for the following considerations.

1) Xilinx is the leading developer of FPGA/CPLD. After they become familiar with the Xilinx design tools, students can use Xilinx FPGA/CPLD in their work after graduation without spending too much time learning to use these tools.
2) Xilinx has an excellent university program supporting web site. This site includes lots of laboratory exercises and student projects collections and links as teaching resources.
3) An excellent step-by-step tutorial plus FAQs are available as Xilinx on-line documents[8]. Extensive collections of FPGA/CPLD application notes are also available on the Xilinx web site.
4) In Xilinx ISE 6.0, digital systems can be designed, synthesized and simulated using schematic capture, VHDL or Verilog hardware description languages. Although 74 series symbols are not available in the ISE version 6.0 library, we have developed many 74series VHDL models for the lab sessions.
5) The license for the logic simulator ModelSim XE starter is free and the XE starter simulator meets the design size for

our lab sessions. The testbench waveform tool is easy to learn and can be used in the test and verification of the digital logic systems in the lab sessions.

All the factors mentioned above play a big part in using Xilinx ISE over Alter Max. The last two factors in particular are the decisive factors.

Lab Design Environment

Introduction to Digital System Design and its laboratory sessions introduce students to the concepts, practices, implementation and verification of digital logic systems. It prepares the students for the on-going digital logic system course, ECE 357 - Intro to VHDL. Xilinx ISE 6.0 version is used in the laboratory exercises, which includes a schematic capture tool (see Figure 1), and Modelsim XE simulator (see Figure 2). Each chapter begins with logic design principles and design examples, followed by a laboratory session where students can build and test the logic circuits.

Xilinx ISE Logic Library

Xilinx maintains software libraries with thousands of functional design elements (primitives and macros) for different device architectures. The catalog of design elements is known as the Unified Libraries. Elements in these libraries are common to all Xilinx device architectures. The primitive and macro logic elements are available in the Unified Libraries for the Xilinx FPGA and CPLD devices. Common logic functions can be implemented with these elements. More complex functions can be built by combining macros and primitives. Several hundred design elements (primitives and macros) are available across multiple device architectures, providing a common base for programmable logic designs. These logic function blocks can be used for schematic design and HDL design.

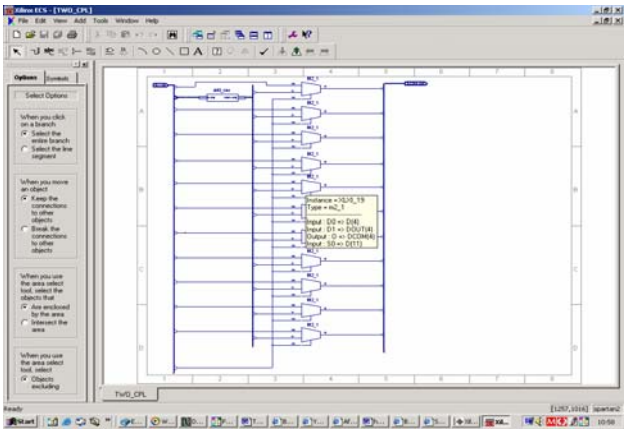

Figure 1. Xilinx ISE Schematic Capture.

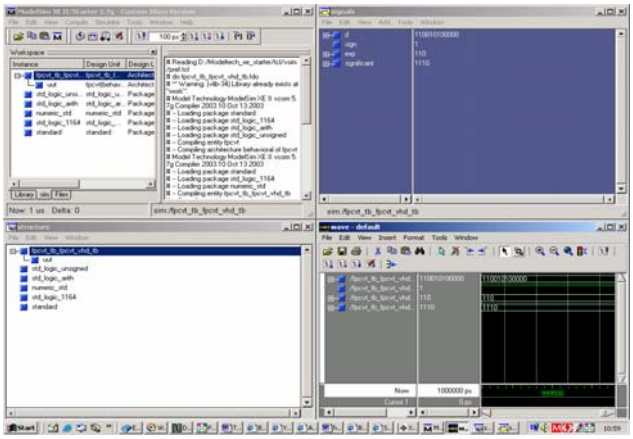

Figure 2. ModelSim XE Logic Simulator.

The following lists are the available functional blocks:

1) Arithmetic functions
2) Buffers
3) Comparators
4) Counters
5) Decoders
6) Flip-flops
7) Latches
8) Logic gates
9) Memory elements
10) Multipliers
11) Shifters and shift registers

In the lab session for the digital system design, students can design, simulate and implement the digital system using available Xilinx symbols. In addition, we have developed many 74series VHDL models which students can use in the lab sessions. Thus, the logic functions of these logic

blocks are first introduced in the textbook and in the lecture, and then students can use these symbols in the lab sessions to get further practice.

Laboratory Exercises and Projects

Laboratory exercises are an important part in the teaching of ECE 270. The objectives of the lab sessions are:

- to help students understand the concepts and principles of the lecture materials
- to give students hands-on experiences with the process of design, implementation, testing and verification of logic systems
- to expose students to the modern EDA design tools for digital system design
- to allow students to learn the debugging and troubleshooting techniques in the design and verification procedures for the digital systems

The first five lab exercises were done on the bread-boards using small-scale integration chips (SSI) that contain only a few logic gates. Students were exposed to the design, verification, debugging and trouble-shooting of the digital logic systems in these lab sessions. Basic electronic equipments such as oscilloscope, digital multi-meters, power supply are used in the first five lab sessions. Students enjoyed these as much as they did the subsequent lab sessions, which involved EDA tools and FPGA/CPLD board verification.

These five lab sessions taught students the basic debugging philosophy described in the following:

1. Work backwards when debugging your circuit
2. Identify the problem, and begin your debugging at the immediate source of the problem.
3. First check the wiring for the IC that produces that output.
4. Use the oscilloscope, digital multi-meter for the troubleshooting.

The techniques and experiences gained from the course will assist students in their future projects such as PCB design and FPGA/CPLD implementation and verification.

The remaining seven lab sessions introduce digital logic blocks using Xilinx ISE tools, such as decoder, encoder, multiplexer, flip-flop, latch, counter, shift register, and sequential state machine. In these lab sessions, students focused on the logic functions and design principles instead of the troubleshooting procedures. The total twelve lab sessions expose the students to a range of design experiences, tool functions and debugging techniques. These typically include the schematics, simulation waveform plots, verification and testing, and a brief lab report.

The following is an outline of the twelve lab sessions:

Lab 1: Introduction to digital devices. The main objective of this lab is to introduce students to simple digital devices and their operations. They also learn the procedure of building simple digital circuits using a digital design kit. In addition, students learned to read the data sheets of digital ICs. This lab helps the students to get first hand experience with good practices of design and debugging. For some students, this is their first experience with the breadboard wiring. Thus, the breadboard connection pictures, digital circuit diagrams, and step by step instructions are provided. Figure 3 is the breadboard connection picture taken from Lab 1 and Figure 4 is the corresponding circuit diagram.

Lab 2: Timing diagram. The main objective of this lab is to learn to use function generator, oscilloscope and digital multi-meters for design, testing and display of the timing diagram of basic logic gates. Students will also know how to draw the timing diagram. For example, Figure 5 is a AND gate circuit diagram used in Lab 2. Students will learn to generate a TTL square wave as an input to the AND gate and the output will be displayed on the oscilloscope.

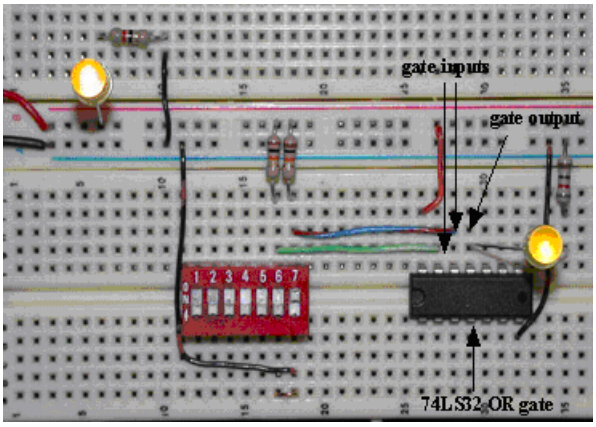

Figure 3. Breadboard Circuit Picture Taken from Lab 1.

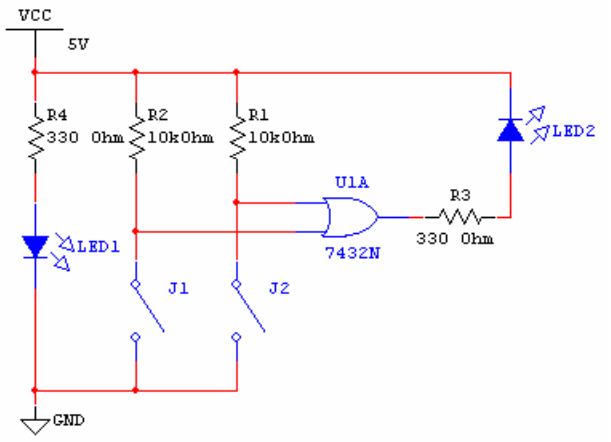

Figure 4. Circuit Diagram for Lab 1.

Lab 3: Implementation of Logic Functions. Students learn how to realize a simple logic function using 74-series components. In this lab, students will design and implement a simple logic function on the breadboard.

Lab 4: Investigation of Timing Hazards. Students learn what causes timing hazards in combinational logic circuits, how hazards due to single input changes can be eliminated, and what happens if more than one input is allowed to change simultaneously. The circuit shown in Figure 6 is used for this lab.

Lab 5: Design of Office Alarm System. In this lab, students learn how to design and solve a problem in the real world and about problem specifications, design, testing and debugging. A simple office alarm system is realized on the breadboard using 74 series logic gates.

Lab 6: Introduction to Xilinx ISE and FPGA/CPLD. The main objective of this laboratory is to teach students the basic of Xilinx ISE tools: design entry, testbench waveform generation, simulation and verification on FPGA boards. The hierarchical design concept using Xilinx ISE is introduced.

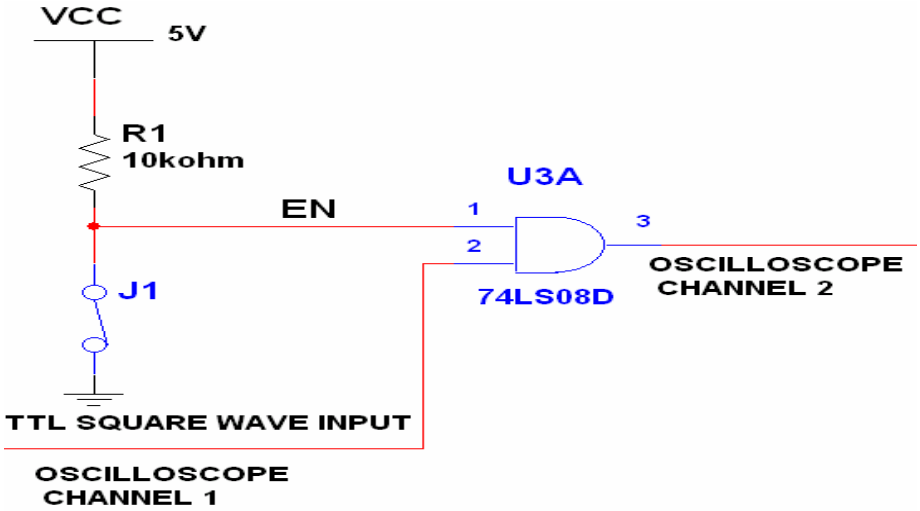

Figure 5. AND Gate Timing Diagram Circuit for Lab 2.

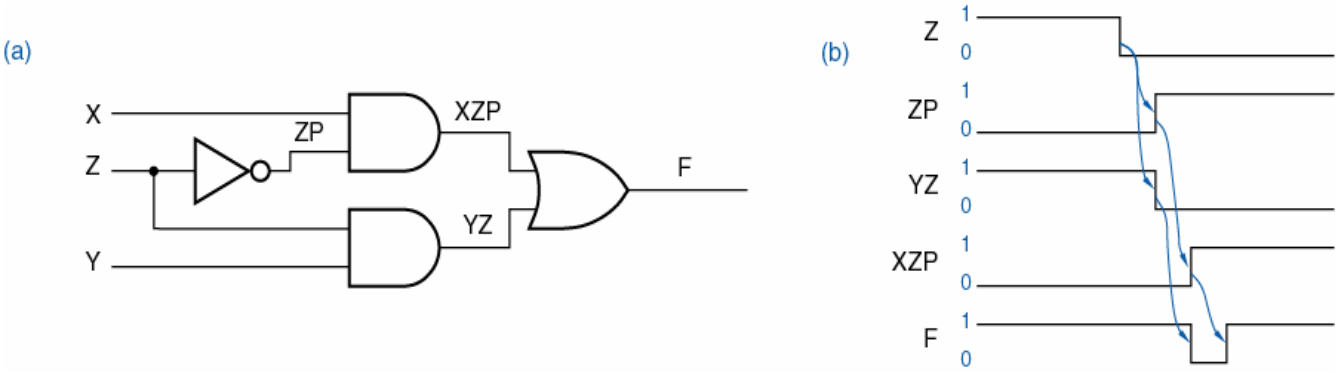

Figure 6. Digital Circuit Used in Lab 4 Timing Hazard.

Lab 7: ALU unit. The main objective is to help students familiar with the logic functions of ALU and use XIlinx ISE tools and FPGA boards for design, simulation and verification. In this Lab, a simple ALU unit is designed using Xilinx ISE schematic capture tools; then, the logic function is simulated, synthesized and downloaded on the FPGA/CPLD boards.

Lab 8: Combinational Logic Block. The main objective of this lab is to let students understand and assimilate the knowledge they learn in the lecture about the basic combinational logic blocks, such as decoder, encoder, multiplexer and the logic block expansions. Students will design a 8-to-1 multiplexer using small size multiplexers. Then a logic function (even number detector) is realized using the multiplexer.

Lab 9: Latches and Flip-Flops. In this lab, students become familiar with the basic operations and functions of SR latch, D latch and D flip-flop. The structure, property and principles of these basic sequential circuits are investigated.

Lab 10: A Simple Traffic Light Controller. In this lab, students learn how to design a sequential state machine and how to draw the state machine using Xilinx StateCAD tools. The light controller system is simulated, implemented and verified on the FPGA board. Figure 7 is the state diagram and Figure 8 shows the simulation waveform.

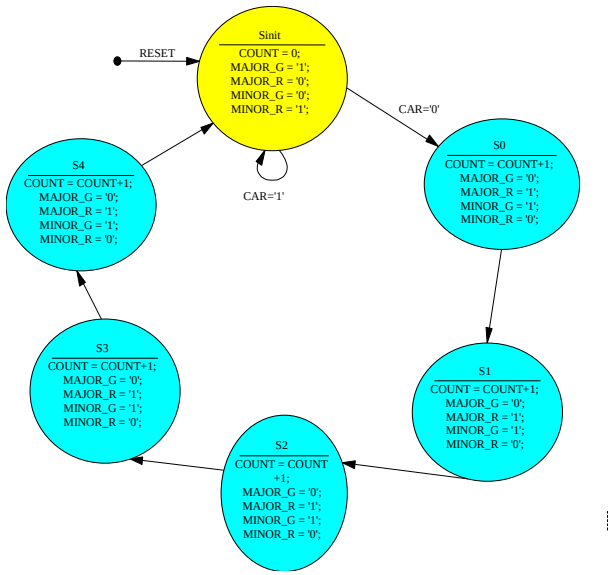

Figure 7. State Diagram for Lab 10 – A Simple Traffic Light Controller.

Lab 11: Asynchronous Counters. Students learn and become familiar with the property of asynchronous counters. Students will design and implement asynchronous module-7 and module-8 counters using Xilinx ISE tools and FPGA boards. Figure 9 shows the logic diagram for an asynchronous module-7 counter.

Lab 12: Synchronous Counter. In this lab, students learn and design synchronous module-7 and module-60 counters using Xilinx ISE tools and FPGA boards. They will learn the differences between asynchronous and synchronous logic circuits. The logic diagram of a synchronous module-7 counter is shown as Figure 10.

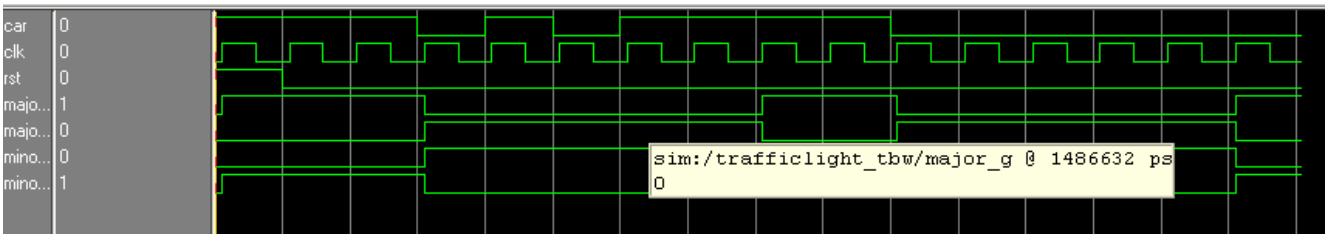

Figure 8. Simulation Waveform for Lab 7.

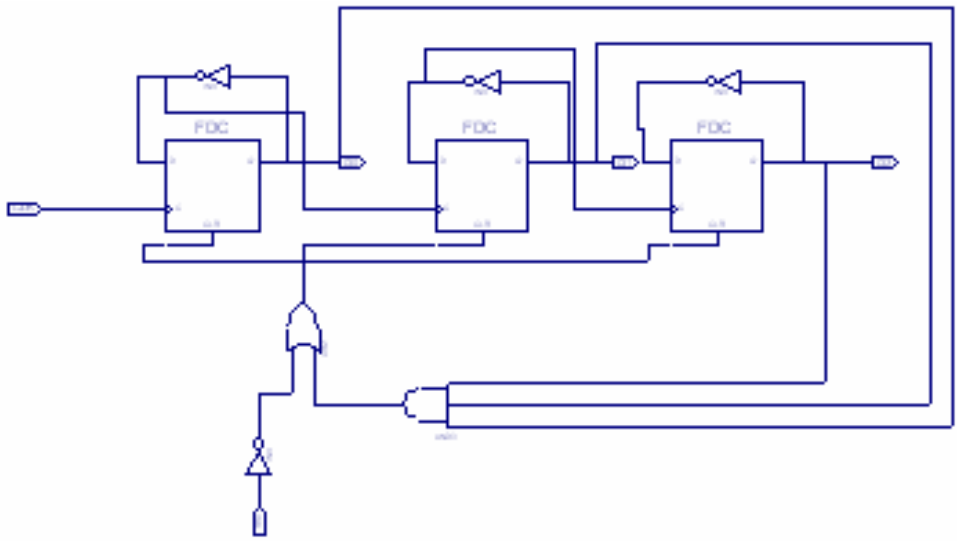

Figure 9. Module-7 Asynchronous Counter for Lab 11.

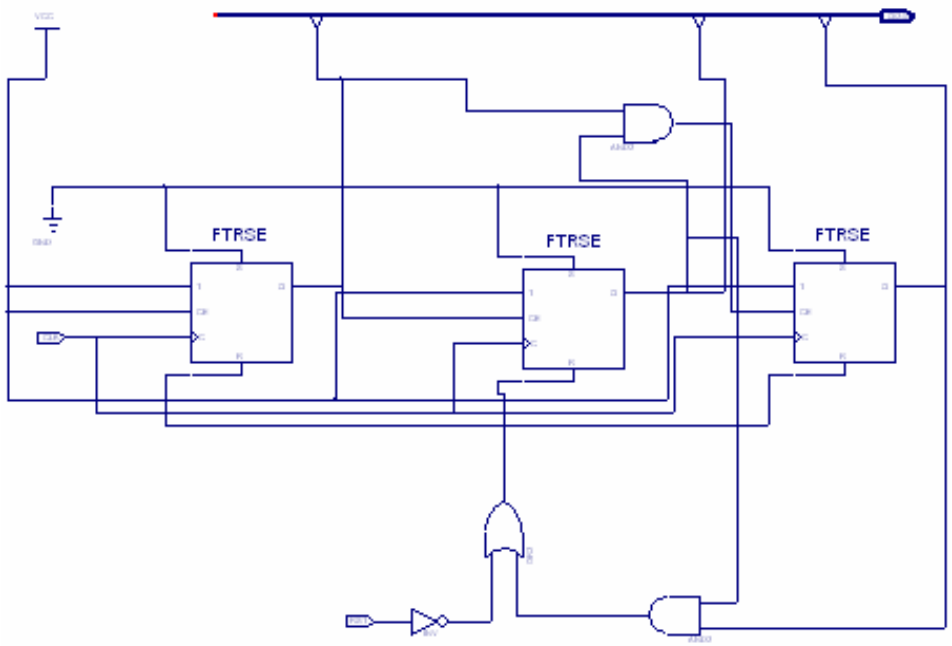

Figure 10. A Synchronous Module-7 Counter.

Computer EDA Applications in the Digital System Design Course

The adoption of EDA tools together with the traditional breadboard wiring lab sessions expose students to both the traditional and state-of-the-art design and implementation experiences. The traditional breadboard lab sessions expose students to the basic design and testing procedure of digital systems. These lab sessions also help students learn some troubleshooting techniques. The last seven lab sessions help them to focus on the principles of the digital systems instead of the debugging and wiring using computer EDA tools. In the first five lab sessions, students spent a significant amount of time on the debugging and troubleshooting. This two-phase lab practice helps students to learn both the hands-on troubleshooting techniques and basic principles of digital systems. Through this approach, students gain a better understanding of the principles and practices of digital logic systems.

Conclusion

Teaching digital systems with bread-board and Xilinx schematic EDA tools and FPGA/CPLD board represents an integrated approach to introducing digital system principles, processes and implementation. It is more effective in the teaching of digital logic systems than the integration using VHDL. This is accomplished by introducing digital design principles, bread-board debugging techniques, design and simulation using EDA tools and verifications on FPGA/CPLD boards. In general, student feedback to this course was very positive. The overall evaluations of this course were 3.50/4.0 (fall 2003), 3.75/4.0 (spring 2004) and 3.6/4.0 (spring 2005). The author also suggests that VHDL be introduced at the junior level after the digital system design is finished. From these teaching experiences in IPFW, the author concludes that integrating EDA and FPGA/CPLD devices instead of the early introduction of HDL can better help students to learn the digital logic systems.

References

1. S. Areibi, "A first course in digital design using VHDL and programmable logic," in *Proceedings - Frontiers in Education Conference*, 2001, vol. 1, pp. 119-123.

2. K. C. Chang, "Including HDL and synthesis in the EE and CSE digital design curriculum," in *Proceedings of the IEEE International Conference on Microelectronic Systems Education, MSE*, 1997, pp. 125-126.

3. M. E. Parten, "Teaching digital design with HDL," in *ASEE Annual Conference Proceedings*, 1997, 7.

4. J. Pedraza-Chavez, D. Baez-Lopez, and J. M. Ramirez, "Use of VHDL as a tool for the teaching of digital systems," in *ASEE Annual Conference Proceedings*, 1995, vol. 2, pp. 2800-2802.

5. C. H. Roth, "Teaching digital system design using VHDL," in *Proceedings - Frontiers in Education Conference*, 1994, pp. 102-105.

6. J. Wakerly, *Digital Design: Principles and Practices*, Prentice Hall, Upper Saddle River, New Jersey, 2001.

7. *"Digital Design: Principles and Practices"* supporting web site, http://www.ddpp.com

8. Xilinx University Program, step by step tutorial, http://www.msu.edu/tutorials.htm

9. Xilinx Library Guide, http://toolbox.xilinx. Com/docsan/xilinx6/books/data/docs/lib/lib 0001_1.html.

Biographical Information

Gupoing Wang is currently an Assistant Professor in the Department of Engineering, Indiana University Purdue University Fort Wayne. He obtained his B.S., M.S., and Ph.D. from Tsinghua University, Nanjing University and the University of Oklahoma respectively in 1988, 1991 and 2003. He teaches courses in digital system design, VLSI Design lab, and Computer Architecture.